

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-153126

(43) 公開日 平成8年(1996)6月11日

(51) Int.Cl.<sup>8</sup>

G 0 6 F 17/50

識別記号

庁内整理番号

F I

技術表示箇所

9191-5H

G 0 6 F 15/ 60

6 5 4 M

審査請求 未請求 請求項の数6 O L (全 6 頁)

(21) 出願番号 特願平6-295238

(22) 出願日 平成6年(1994)11月29日

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72) 発明者 大鹿 緑

神奈川県川崎市幸区柳町70番地 株式会社

東芝柳町工場内

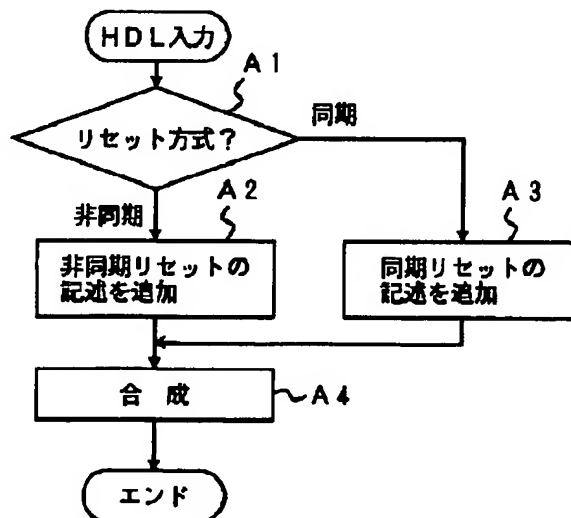
(74) 代理人 弁理士 鈴江 武彦

(54) 【発明の名称】 回路設計支援装置及び方法

(57) 【要約】

【目的】 回路設計における冗長作業の繰返しを削減して開発時間を短縮する。

【構成】 ハードウェア記述言語 (HDL) で記述したレジスタトランスファレベル (RTL) の回路情報に対してリセット方式が同期か非同期かを選択設定することで (ステップ A 1)、その設定内容に応じて上記回路情報へのリセットの記述を自動付加する (ステップ A 2、A 3)。



【特許請求の範囲】

【請求項1】 ハードウェア記述言語で記述したレジスタトランスファレベルの回路情報に対し、リセット方式が同期か非同期かを設定する設定手段と、この設定手段の設定内容に応じて上記回路情報へのリセットの記述を自動付加する記述付加手段とを備えたことを特徴とする回路設計支援装置。

【請求項2】 ゲートレベルで記述された回路情報に対し、リセット方式が同期か非同期かを設定する設定手段と、この設定手段の設定内容に応じて上記回路情報を同期リセット方式あるいは非同期リセット方式の同回路情報に自動置換する記述置換手段とを備えたことを特徴とする回路設計支援装置。

【請求項3】 上記記述置換手段は同期リセット方式の回路情報への置換に代えてリセット信号との論理積をとるゲートの回路情報を付加することを特徴とする請求項2記載の回路設計支援装置。

【請求項4】 ハードウェア記述言語で記述したレジスタトランスファレベルの回路情報に対し、リセット方式が同期か非同期かを設定する設定処理と、この設定処理の設定内容に応じて上記回路情報へのリセットの記述を自動付加する記述付加処理とを有することを特徴とする回路設計支援方法。

【請求項5】 ゲートレベルで記述された回路情報に対し、リセット方式が同期か非同期かを設定する設定処理と、この設定処理の設定内容に応じて上記回路情報を同期リセット方式あるいは非同期リセット方式の同回路情報に自動置換する記述置換処理とを有することを特徴とする回路設計支援方法。

【請求項6】 上記記述置換処理は同期リセット方式の回路情報への置換に代えてリセット信号との論理積をとるゲートの回路情報を付加することを特徴とする請求項5記載の回路設計支援方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、ASIC (Application Specific Integrated Circuit: 特定用途向けIC) 等の回路設計を行なうCADに用いられる回路設計支援装置及び方法に関する。

【0002】

【従来の技術】 ASIC等の回路設計を行なうCADの環境下にあつては、通常、図面での入力あるいはハードウェア記述言語での入力のいずれの場合でも、予めリセット回路を意識したセルの選択及び記述を行なうようになっていた。

【0003】 すなわち、図面入力のようにゲートレベルの設計では、リセット方式が同期であるか非同期である

かによって使用するレジスタのセルや接続が異なっている。同様に、ハードウェア記述言語（以下「HDL」と略称する）を使用したレジスタトランスファレベル（以下「RTL」と略称する）の設計でも、リセット方式が同期であるか非同期であるかによって記述が異なっている。

【0004】

【発明が解決しようとする課題】 一般に回路設計では、同一モジュール内でリセット方式が同期、非同期のレジスタを混在させることは極めて希であり、同期か非同期かのリセット方式が決まれば、RTLの入力であれ、ゲートレベルの入力であれ、リセット回路の記述は回路の機能に関係なく一様に決定し得るものである。

【0005】 しかしながら、リセット機能の記述やリセット付きレジスタのセルを選択することは必須であり、同一記述やリセットのソースとセルのリセットピンを接続するような冗長な操作を繰返し行なわなければならない。

【0006】 また、特に同期リセット方式で、リセット信号とレジスタの入力データとの論理積をとるゲートの挿入を行なっている場合、論理合成及び最適化を行なうことにより、リセット信号とレジスタの入力データとの論理積をとるゲートとレジスタの入力の間に、複数の組合せ回路が挿入され、その結果として時にレジスタのクリアが正常に行なわれなくなってしまうという問題が発生していた。

【0007】 本発明は上記のような実情に鑑みてなされたもので、その目的とするところは、回路設計における冗長作業の繰返しを削減することで開発時間を短縮し、且つリセット方式による動作の保証された回路を得ることが可能な回路設計支援装置及び方法を提供することにある。

【0008】

【課題を解決するための手段】 すなわち本発明は、ハードウェア記述言語で記述したRTLの回路情報に対してリセット方式が同期か非同期かを設定することで、その設定内容に応じて上記回路情報へのリセットの記述を自動付加し、あるいはゲートレベルで記述された回路情報に対してリセット方式が同期か非同期かを設定することで、その設定内容に応じて上記回路情報を同期リセット方式あるいは非同期リセット方式の同回路情報に自動置換するようにしたものである。

【0009】

【作用】 上記のような手段をとることにより、RTL及びゲートレベルのいずれで回路設計を行なう場合であっても、リセット方式に関する記述やリセットのソースとセルのリセットピンを接続するような冗長な操作を削減して開発時間を短縮することができる。

【0010】

【実施例】 以下図面を参照して本発明の一実施例を説明

する。図1はワークステーションによりCADシステムを構築した場合の回路構成を示すもので、SBはシステムバス、11はマイクロプロセッサ(CPU)、12は同マイクロプロセッサ11によりアクセスされるRAM、13は同ROMである。

【0011】マイクロプロセッサ(以下CPUと称す)11は装置全体の制御を行なうもので、上記RAM12及びROM13をアクセスして、入力指示に従うプログラムの起動で各種回路設計の処理等を実行する。

【0012】RAM12は回路設計時のレベルで記述された各回路情報その他データ処理上の各種データを記憶する領域からなり、ROM13はオペレーティングシステム等のプログラムや各所固定情報を格納している。

【0013】また、14はキーボード(KB)、16はポインティングデバイスとしてのマウス、17は表示部であるCRT、20はハードディスク装置(HDD)、22はフロッピーディスク装置(FDD)、24はインタフェース装置(IF)である。

【0014】キーボード14は、回路設計等に必要の入力情報をキーボードインタフェース(KB-IF)15を介してCPU11へ入力するもので、このキーボードインタフェース15は上記マウス16でのポインティング入力の動作制御も兼ねている。

```
「if (CLK) ' event and CLK=' 1' ) then
  Q <= D;
end if;」
```

なる記述で示している。

【0020】この図4(1)で示したような回路に関する記述が複数なされている状態で、これを一括してリセット方式を選択設定するべく、RTLでリセット方式を付加する記述をキーボード14あるいはマウス16により入力すると、図2の処理によりまずそのリセット方式の内容、すなわちクロックの同期リセットであるか、非同期リセットであるかを自動で選択判断する(ステップA1)。

【0021】ここで非同期リセットであると判断した場合、上記図4(1)に示したF/F回路に関する記述に、クロックに関係なくリセット信号が入力された時点でレジスタがクリアされるような非同期リセット方式に関する記述を図4(2)中に下線で示すように自動追加する(ステップA2)。

【0022】すなわち図4(2)では、当該F/F回路が非同期リセットであることを示す記述

```
「if (RESET=' 0' then
  Q <= ' 0' ;
```

を上記F/F回路の記述の前に追加挿入すると共に、当該F/F回路がクロック及びリセット信号の入力に直接基づいた処理工程をとることを示す

```
「process (CLK, RESET) 」
```

【0015】ハードディスク装置20は、ディスクインタフェース(HDD-IF)21を介してCPU11の制御の下に、回路設計に伴うHDLの変換プログラムやRTLあるいはゲートレベルで記述された回路情報、上記変換に必要なライブラリ等を記憶保存する。

【0016】同様にフロッピーディスク装置22は、ディスクインタフェース(FDD-IF)23を介してCPU11の制御の下に、上記ハードディスク装置20と同等の各種情報を記憶保存する。

【0017】インタフェース装置24は、CPU11の制御の下にネットワーク接続された図示しない他のワークステーションとの間でデータの送受を行なう。次に上記実施例の具体的な動作について説明する。

【0018】図2はHDLを使用したRTLでの回路設計時における、リセット方式の設定処理内容を示すものであり、図4はVHDL(VHSIC Hardware Description Language)を用いた上記RTLでの記述例を示す。

【0019】すなわち図4(1)は、RTLでリセット方式を意識せずに動作、機能を記述した1回路として、クロックの立上がりで入力データをラッチするリセットなしのフリップフロップ(F/F)回路を図中の太字で示した

なる記述に書換え、さらに上記F/F回路の記述の後に「end if;」

なる記述を追加挿入するものである。

【0023】また、上記ステップA1で同期リセットであると判断した場合は、上記図4(1)に示したF/F回路に関する記述に、クロックに同期してレジスタがリセットされる同期リセット方式に関する記述を図4

(3)中に下線で示すように自動追加する(ステップA3)。

【0024】すなわち図4(3)では、当該F/F回路が同期リセットであることを示す記述

```
「if (RESET=' 0' then
  Q <= ' 0' ;
```

を上記F/F回路の記述中

```
「if (RESET=' 0' then」
```

の次に追加挿入すると共に、このF/F回路の記述の後に

```
「end if;」
```

なる記述を追加挿入するものである。

【0025】上記ステップA2あるいはステップA3のリセット方式の記述追加処理後、変換プログラムによってゲートレベルでの論理合成及び最適化を実行し(ステップA4)、以上でこの図2の処理を終了するものであ